AMENDMENTS TO THE DRAWINGS

The attached sheet of drawings include changes to Fig. 4. This sheet, which includes Fig. 4, replaces the original sheet including this figure. In Fig. 4, previously omitted labels "44b", "44c" and "44d" have been added.

Attachment: Replacement Sheet(s)
Annotated Sheet Showing Changes

REMARKS/ARGUMENTS

Reconsideration and allowance of this application are respectfully requested. Currently, claims 2-20, 29 and 31-37 are pending in this application.

Rejection Under 35 U.S.C. §112:

Claims 1-20 were rejected under 35 U.S.C. §112, second paragraph, as allegedly being indefinite. Applicant submits that still pending claims 2-20 are in full conformance with 35 U.S.C. §112, second paragraph. Applicant therefore respectfully requests that the rejection under 35 U.S.C. §112, second paragraph, be withdrawn.

Rejections Under 35 U.S.C. §103:

Claims 1-4, 6, 11-12, 29 and 30-32 were rejected under 35 U.S.C. §103 as allegedly being unpatentable over Applicant admitted prior art in view of Aridor. Applicant respectfully traverses this rejection.

In order to establish a prima facie case of obviousness, all of the claim limitations must be taught or suggested by the prior art. The combination of the admitted prior art and Aridor fails to teach or suggest all of the claim limitations. For example, the combination fails to teach or suggest "said at least one second computer thereby being programmed to receive said data defining the plurality of heterogeneous programs and said data defining said co-ordinating program, and to execute, in parallel, said co-ordinating program and said heterogeneous programs," as required by new independent claim 36 and its dependents. The combination also fails to teach or suggest the following limitations required by new independent claim 37 and its dependents:

> "receiving in the at least one second computer the co-ordinating program and the plurality of heterogeneous programs and

> executing in parallel the co-ordinating program and the heterogeneous programs;
>
> wherein the execution of the co-ordinating program co-ordinates the operations of the heterogeneous programs on the second computer...."

Moreover, the combination also fails to teach or suggest supplying a plurality of parallel processing task programs and a co-ordinating program from a first computer to a second computer, and co-ordinating operation of the task programs through the co-ordinating program on the second computer, as required by independent claim 29 and its dependents.

The above limitations are supported by, for example, the exemplary embodiment illustrated in Fig. 4. In Fig. 4, an agent control program 40 is provided on a user computer 10. The agent control program 40 is in communication with a co-ordinating agent program 42 resident on one of a plurality of remote computers (e.g., computer 30a in Fig. 4). A plurality of task agent programs 44a-44d are also located on computer 30a. Each task agent program 44a-44d is arranged to communicate with co-ordinating agent program 42. Task agent programs 44a-44d operate in parallel (see page 6, line 4 of the originally-filed specification). A scout agent program 46 is provided on computer 30b, another one of the remote computers. The co-ordinating agent program 42 communicates locally with task agent programs 44a-44d, and with agent control program 40 on the user computer 10, and with the scout program 46 located on the other remote computer 30b. The co-ordinating agent program 42 decides if and when it should move from the remote computer 30a to the other remote computer 30b to re-commence execution on the other remote computer. (See page 6, line 16 to page 7, line 8 of the originally-filed specification).

The agent control program 40 creates code for task agents, determines which remote computers are suitable for performing the task, and in what order to create co-ordinating agent program 42. The co-ordinating agent program 42 is sent to the first remote agent computer 30a together with code to enable the creation of task agents on the first remote computer 30a. When the resources of the first remote computer 30a cease to be suitable, the co-ordinating agent program removes itself either based on the itinerary determined by the user agent, or from the reports of the scout agents, and moves itself and the task agents 44 to the remote computer 30b via the telecommunications network.

Aridor relates to agent design "patterns." In this context, a "template" for a design seeks to provide a structure for specific problems/solutions which are then addressed by designing agents using the "pattern". By definition, a (design) pattern is a formulization of a problem/solution pair to make an object-oriented design decision. A pattern codifies existing design knowledge so that developers don't unnecessarily have to repeat development of these designs.

In Aridor, patterns are conceptually divided into separate task and interaction classes. One type of task pattern is the "plan" task pattern, which provides a way of defining the co-ordination of multiple tasks to be performed on multiple hosts. More specifically, the "plan" task pattern adopts a workflow concept to organize multiple tasks to be performed in sequence or in parallel by multiple agents. The plan encapsulates the task flow, which is then hidden from the agent, and fixes the behavior of the agents at the design time. Thus the "plan" task pattern remains at the host computer - it does not migrate to remote computers. This is consistent with the "Master-Slave" task pattern Aridor describes, which allows a master agent to delegate a task to a slave agent, which

then moves to the destination host, performs the assigned task, and then returns with the possible result of that task. The Master agent is aware of the plan, not the task agent.

Aridor considers how agents at different destinations may need to interact **locally** between themselves. Aridor teaches that when agents are dispatched from their origins to a central destination where they interact locally amongst themselves, a problem with synchronization occurs as the agents will initially be at different hosts. In Aridor, a "meeting pattern" is used to encapsulate a specific destination (meeting place) and a unique identifier. In general, agents that need to locally interact are equipped with a meeting object. Indeed, the "meeting" interaction pattern is described as "Provides a way for two or more agents to initiate **local** interaction at a given host (emphasis added)." (See page 109, col. 2). Each agent then dispatches itself independently to the meeting place, where it uses the unique identifier to locate a specific local meeting manager object to register itself. The meeting manager object (which is inherently at the local host already) then notifies already registered agents of the new arrival, so that interactions can take place amongst themselves.

In Aridor, agents thus interact locally at a pre-determined, specified location, at which a local meeting manager is provided to facilitate agent interaction at the specified location. Aridor does not describe dispatching the "meeting manager" to the specified location with each team of "agents", as in Aridor agents from a plurality of origins interact via a meeting manager.

In contrast to Aridor where a meeting manager must be already provided at a remote computer to coordinate local interactions amongst agents, exemplary embodiments of the present invention migrate both task agents and a co-ordination

program to a remote host, in dependence for example on either the original itinerary generated by the user computer or on the reports of local scout agents. Nothing in Aridor teaches generating a "meeting manager" which facilitates meetings on one computer where a plurality of agents perform a first set of tasks, and then migrating this program on to further computers together with said plurality of task agents.

Applicant thus submits that independent claims 29 and 36-37 are not "obvious" under 35 U.S.C. §103 over AAPA and Aridor. The advantages of providing a co-ordinating program with the task agents is that the capabilities of the remote computers can be assessed for the task itself (i.e., coordinating agent and task agents), and a support program provided when necessary. This is not possible in the system contemplated by the combination of AAPA and Aridor.

Accordingly, Applicant respectfully submits that claims 1-4, 11-12 and 29 are not "obvious" over Applicant admitted prior art and Aridor and respectfully requests that the rejection of these claims under 35 U.S.C. §103 be withdrawn.

Claims 13 and 35 were rejected under 35 U.S.C. §103 as allegedly being unpatentable over Applicant admitted prior art in view of Aridor and further in view of Berghoff et al (hereinafter "Berghoff"). Claims 5, 7, 8, 14-20 and 33-34 were rejected under 35 U.S.C. §103 as allegedly being unpatentable over Applicant admitted prior art in view of Aridor and further in view of Kozuka (U.S. '394). Claims 9-10 were rejected under 35 U.S.C. §103 as allegedly being unpatentable over Applicant admitted prior art in view of Aridor and further in view of "Objectspace." Applicant respectfully traverses these rejections. Neither Berghoff, Kozuka nor "Objectspace" remedies the above described deficiencies of Applicant admitted prior art and Aridor with respect to base

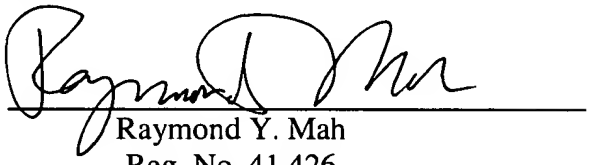
claim 36 or 37. Applicant thus requests that these rejections under 35 U.S.C. §103 be withdrawn.

Conclusion:

Applicant believes that this entire application is in condition for allowance and respectfully requests a notice to this effect. If the Examiner has any questions or believes that an interview would further prosecution of this application, the Examiner is invited to telephone the undersigned.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: 
Raymond Y. Mah
Reg. No. 41,426

RYM:sl
901 North Glebe Road, 11th Floor
Arlington, VA 22203
Telephone: (703) 816-4044
Facsimile: (703) 816-4100

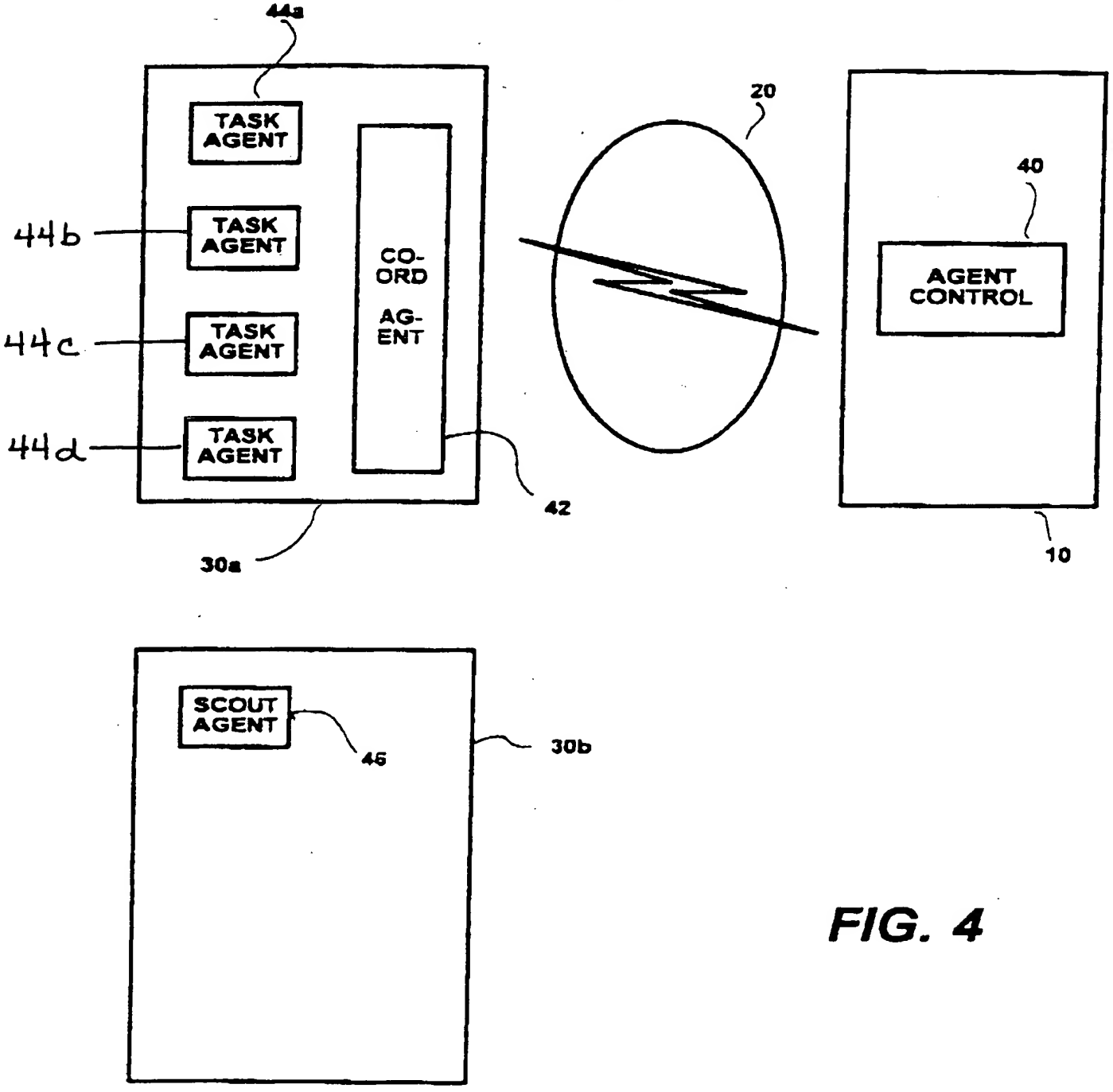


FIG. 4